Chapter 4

HOPLP-MUL: Link Prediction in Multiplex Networks based on Higher Order Paths and Layer Fusion

The previous chapter attempted to exploit edge relevance quantification using ego networks to create a quasi-local similarity-based link prediction method for simple networks. *ELP* aggregates different Ego regions of nodes to quantify edge relevance. This process can be visualized for a small graph by aggregating several graphs with the same node set with different edge weights. A correlation to multiplex networks with similar node sets in different layers of edge sets becomes evident. There exists an opportunity to apply edge relevance quantification for quasi-local link prediction in multiplex networks. This chapter [1] proposes a novel link prediction method for multiplex networks called $HOPLP-MUL$ (Higher Order Path-based Link Prediction for Multiplex Networks). Multiple kinds of connections (links) may be encoded into distinct layers in multiplex networks, with each layer representing a particular type of link. Even if the type of linkages in various layers varies, the nodes themselves and their underlying relationships are retained. Considering the combined structure of all the layers, we can

[1] Published in Applied Intelligence, DOI:https://doi.org/10.1007/s10489-022-03733-8

achieve a complete overview of the network, which is impossible to achieve using any single layer itself. In this chapter, we theorize that this summarized graph (overview) provides us with an opportunity to determine the regional influence of nodes with greater certainty, and we can exploit this for more accurate link prediction.

4.1 Introduction

Though the majority of research in the field of link prediction is concentrated on single-layer networks, real-world interactions are usually too complex to be represented as such simple networks. Online networking platform LinkedIn provides an interesting use case for such multi dimensional user behaviour. The users themselves can be considered as nodes of a social network graph, and the attributes of nodes such as connections between followers and followees, shared interest as well as other degrees of interactions can be considered as different kinds of links between nodes. This social network can be represented as a multi-layered multiplex network. The nodes in all layers would represent the users themselves while a connection in a particular layer would represent a connection between users based on the type of interactions represented by the layer itself. For the most accurate link prediction in multiplex networks, any proposed solution should make use of information of all layers to estimate the possibility of link in a single layer as the users themselves remain the same.

In this chapter a link prediction method on multiplex networks is proposed, based on an iterative calculation of link similarities on higher-order paths. The primary motivation behind $HOPLP-MUL$ method is to use higher-order paths to better estimate neighborhood similarities with the understanding that node influence across three degrees of influence [29] is taken into account. To accurately gauge the information represented by different layers of the multiplex graph, a density-based summarization model is used. The inspiration for the proposed method comes from quasi-local similarity based methods which attempt to incorporate different types of information

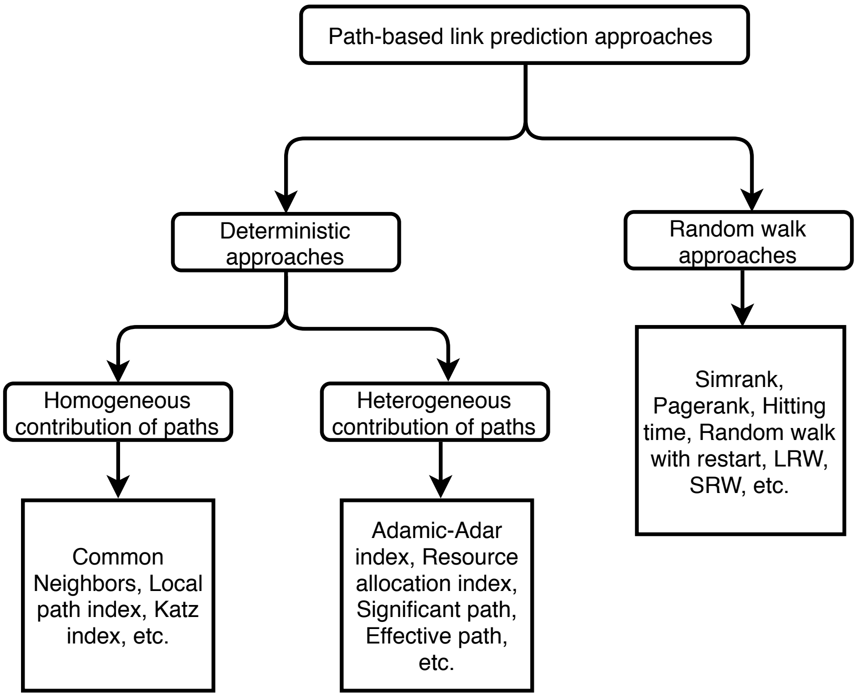

Path-based link prediction approaches
Deterministic approaches
Random walk approaches
Homogeneous contribution of paths
Heterogeneous contribution of paths
Simrank, Pagerank, Hitting time, Random walk with restart, LRW, SRW, etc.
Common Neighbors, Local path index, Katz index, etc.
Adamic-Adar index, Resource allocation index, Significant path, Effective path, etc.

FIGURE 4.1: Taxonomy of Path-based approaches to Link Prediction (Ajay et al.[1])

(both local and global) into a related index for link prediction. Such quasi-local methods have shown improved link prediction performance in case of simple networks when compared to methods based on purely local and global features. Based on the 3 Degree of Influence Phenomenon by Christakis and Fowler [28], it is believed that to calculate the likelihood for a link between nodes, it would be better to take the overall region of influence of nodes into account instead of just the immediate neighbors. Hence, a method to summarize and process all the information spread across layers into one concise weighted graph (compression) is utilized. Then link prediction is performed on this graph using longer length paths and calculate link probabilities on specific layers accounting for differences in layer densities (decompression).

4.1.1 Path Based Approaches to Link Prediction

Considering the nature of the actual path used by path-based approaches to link prediction, existing research can be broadly classified into the following categories, as

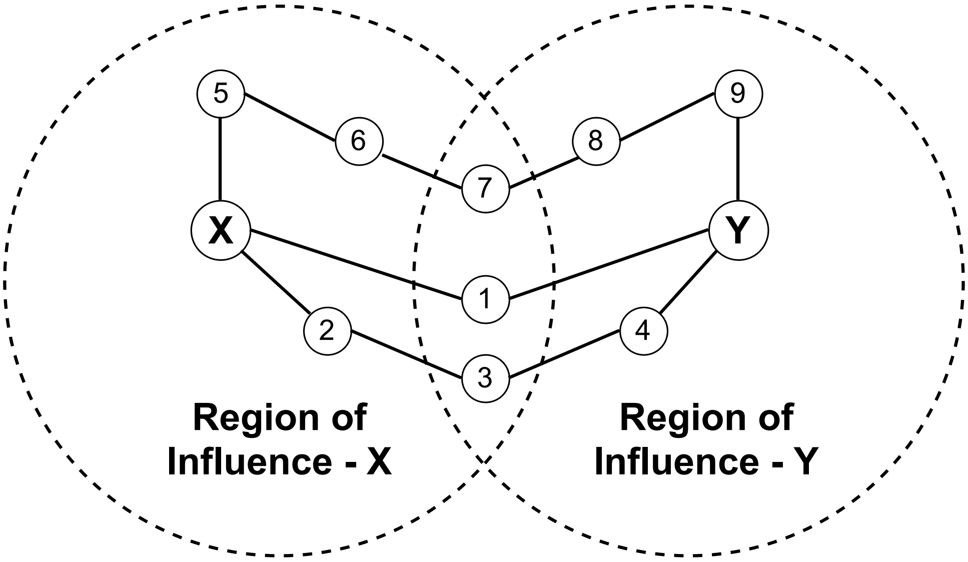


FIGURE 4.2: HOPLP-MUL Concept Graph showing regions of influence of nodes *X&Y* with intersections between them which denote possible information flow paths.

shown in Fig.4.1. Fundamental differences based on the nature of paths can be observed using either a deterministic or a random walk-based approach. Paths used by the deterministic method have fixed starting and ending vertices, and all the paths between these points are usually explored to get a better overall estimate of network structure. However, this approach is more time-consuming than the random walk-based approach. In the random walk-based approach, a fixed central node is usually used as a starting point, and random walks from this node are performed to get a better estimate of the neighborhood of this node. A corpus is created which keeps track of uniqueness and frequency neighborhood node occurrences. At the time of link prediction, this corpus of nodes is compared to estimate link likelihoods. If the sparseness of a given graph is considered and the parameters of a random walk are adequately tuned, the paths considered by the deterministic and random walk approaches may become the same.

Within the deterministic class of approaches, another classification level can be made based on the contribution of paths, i.e., either homogeneous or heterogeneous, to the similarity score computations (Fig.4.1). While homogeneous methods consider the contribution of all paths to be the same, heterogeneous methods have a scheme for prioritization of such paths based on some parameter, for example, length of path or degrees of intermediate nodes. In a previous work, SHOPI [1], a path-based link prediction approach is proposed for simple unweighted networks based on the assumption that similar nodes can have a maximum of six degrees of separation between them. All degrees from one right to six should be considered for similarity calculations, albeit with different priorities. However, those assumptions do not hold for multiplex networks, particularly in the setting of this chapter where link prediction has to be performed on a summarized weighted graph, and those results have to be extrapolated onto all layers of the multiplex network. Also, the mechanism for influence transfer was not explored in detail. In this chapter, different regions of influence and the nature of influence transfer are explored to provide the best possible solution to the link prediction problem in the context of multiplex networks. The method, $HOPLP-MUL$, can be considered a deterministic heterogeneous method as we consider all paths, and their length defines the relative importance of paths to similarity score computations. This length-based dampening results in higher penalization of longer paths when compared to their shorter counterparts leading to better estimation of overall link likelihoods.

4.1.2 HOPLP-MUL Concept

The concept behind $HOPLP-MUL$ approach is demonstrated using Fig.4.2. In this graph, the task to find link prediction probability between nodes $X\&Y$. It can be clearly observed that if the region of influence of nodes is assumed to be one hop from the node of origin, then the only relevant path would be $X-1-Y$. If it is assumed that the region of influence of a node extends to two hops from its point of origin then the relevant paths considered for link prediction would be both $X-1-Y$ and $X-2-3-4-Y$. Finally using the 3 Degree of Influence phenomenon given by Christakis et al. [28, 29], it can be assumed that the region of influence of a node extends to 3 hops away from it. Using this assumption, it can be seen that for calculation of link probability between $X\&Y$, we have to consider the smaller paths ($X-1-Y$ and $X-2-3-4-Y$) as well as the longest possible path under the 3 Degree of Influence constraint, i.e., $X-5-6-7-8-9-Y$. In $HOPLP-MUL$, all these paths are used for more accurate link prediction.

4.2 Proposed Work

Based on existing research on real-world networks, it is evident that social networks show topological properties, which can be attributed to three significant phenomena, i.e., small-world association [152–156], clustering [157], and scale-free behavior [158]. These phenomena are associated with the features of the path, clustering coefficient, and degree distribution of the network. In $HOPLP-MUL$ approach, the path property of the network is utilized to estimate better link likelihoods in a weighted summarized single-layer graph created from multi-layer or multiplex-networks. These likelihood scores are converted to link probabilities on the original multiplex network in a layer-specific manner. It is assumed that a sender (start node) sends information to a receiver (end node) using these paths, and we attempt to quantize and sum up the effect of this information flow on the relevant paths. The process is somewhat similar to the resource allocation index [132], but it only considers two-length paths while $HOPLP-MUL$ proposal aims to evaluate the effect of higher-order (longer) paths. This combined information flow estimates for link likelihood between the start and end nodes. The initial two path length score is calculated by just taking familiar neighbors into account without including any degree of penalization in the process.

4.2.1 HOPLP-MUL Framework

The proposed algorithm, $HOPLP-MUL$, consists of three basic steps -

- The first step is collating the disparate information from all levels into a single summary weighted graph. This is accomplished by altering an aggregation modelling approach for changes in the total edge densities of layers, as discussed in Section 4.2.2.

- Secondly, the initial significance of immediate neighbors is calculated on possible links. To better evaluate the working of $HOPLP - MUL$ algorithm, two possible methods to estimate the significance of path-length 2 are proposed. One is based directly on the resource allocation index, and the other is a modification of resource allocation index, taking into account the power-law effect of total edge weights, as discussed in Section 4.2.3.
- The third step is the iterative step, where the significance of longer paths is calculated by taking the information of their components into account. For instance, to calculate the likelihoods for path length x, likelihoods for path length $x-1$ and the dampened influence created after adding an edge to the path of length $x-1$ are used, as discussed in Section 4.2.5.

4.2.2 Network Summarization

Several types of summarization and representations have been used for multiplex networks in literature, some of the popular ones are boolean operator-based [159] and embedding-based [160, 161]. Boolean operator-based [159] summarization have not been used because the resulting graph does not offer the property of edge weighting based on layer densities and treats all edges across all layers to be the same. Weights were used to express dissimilarities in density across layers in this technique, and weights also aid in the subtle transfer of edge probabilities from the summary graph to real layers. In suggested solutions to the link prediction issue in multiplex networks, Boolean-based summarization approaches have been utilised, particularly in circumstances where the link prediction problem in specific layers is characterised as a multiple attribute decision making (MADM) problem [31, 116]. The fundamental limitation of these approaches is that for link prediction in each layer, a new initial information matrix has to be defined, which contains similarity of edges of the current layer and occurrence information of same edges in other layers. This matrix is then processed for calculating prediction scores for possible edges. In $HOPLP - MUL$ proposed approach, the summarized weighted graph on which link prediction is performed remains the same irrespective of the layer on which link prediction is performed. Hence link prediction can be done on all layers after just one round of processing the summarized weighted graph.

Embedding-based [160, 161] summarization was not used because it would add two distinct avenues of increased complexity to this framework. First, we would have to calculate weights (similarities) of edges for all layers before link prediction and also at the time of transposing probabilities to original layers. For example, if three similarity-based methods are used to generate scores for edges on a 3-layered multiplex network, this would involve running similarity calculations nine times. These similarity-based embeddings can be used directly for classification tasks that model link prediction as a supervised machine learning problem [118]. Another option is using random walks for generating embeddings and then using these embeddings for edge classification. However, these embeddings by themselves have a higher complexity for generation than similarity-based methods, even when edge embeddings are directly generated from node embeddings. Methods which generate embeddings taking into account information from all layers are complex because layers of a multiplex network can vary widely in properties such as average density and clustering coefficients.

In $HOPLP - MUL$ concept, network summarization is the process of converting many interaction (multiplex) networks to a single weighted network. To construct this weighted network, a topological integration technique was employed. Thus, using Equation 4.1 [162], one can calculate the connection strength $A_M(n_1, n_2)$ of any existing edge (n_1, n_2) in such a network. The graph G_M is obtained by combining all edges with some degree of connectivity into a single graph.

$$A_M(n_1,n_2) \leftarrow \frac{1}{n}\sum_{j=1}^{n}\{A_j \| A_j = [a^j_{n_1,n_2}]_{\|V\|\times\|V\|}\} \tag{4.1}$$

where,

$$a^j_{n_1,n_2} \leftarrow \begin{cases} 1 & \mathit{if}\ \exists(n_1,n_2) \in E_j, j \in [1,n] \\ 0 & \mathit{otherwise} \end{cases}$$

In $HOPLP-MUL$, a significantly modified strategy is offered that takes into account the proportional densities of the layers. By combining this with an appropriate strategy for re-transforming summary graph probabilities to the original layers, it is our belief that more accurate findings would be attained. Equations 4.2 and 4.3 describe the two suggested parameters for layer fusion (compression) and likelihood transposition (decompression).

$$CZ(j) \leftarrow \frac{1}{\|E_j\|} \tag{4.2}$$

$$DCZ(j) \leftarrow \frac{\|E_{G_{HOPLP}}\| - \|E_j\|}{\|E_{G_{HOPLP}}\|} \tag{4.3}$$

The graph G_{HOPLP} will be the same as G_M, where nodes in these graphs have an edge if any of the layers have the same edge. The modified summarized weight matrix is as follows -

$$A_{HOPLP}(n_1,n_2) \leftarrow \frac{1}{n}\sum_{j=1}^{n}\Big(\{A_j \| A_j = [a^j_{n_1,n_2}]_{\|V\|\times\|V\|}\} \\ * CZ(j)\Big)$$

4.2.3 Initial Significance of Path Length 2

Using the social paradigm of the significance of familiar neighbors based on degree, two variations of calculating the initial significance are proposed which are used as input to calculate further significance using longer paths. One is based on just the resource allocation index (Equation 4.4), and the other is based on a modification of resource allocation index, taking into account power-law effect of total edge weights (Equation 4.5), which is equivalent to the Adamic-Adar index for weighted graphs. Instead of the degree of a node representing its significance, a value calculated by summing up all the weights of edges of the node is used (from matrix A_{HOPLP}).

$$IS(n_1,n_2) \leftarrow \sum_{cn\in N(n_1)\cap N(n_2)} \left(\frac{1}{\sum_{x\in N(cn)} w[x,cn]} \right) \tag{4.4}$$

$$IS(n_1,n_2) \leftarrow \sum_{cn\in N(n_1)\cap N(n_2)} \left(\frac{1}{\log(\sum_{x\in N(cn)} w[x,cn])} \right) \tag{4.5}$$

4.2.4 Dampening Function

The basic premise behind $HOPLP-MUL$ approach is that longer paths between nodes also play a relevant role in predicting possible edges between them. Though the region of influence of a single node can stretch three hops away from it (3 Degree Phenomenon), it is evident that node influence decreases as we move away from it. Hence a dampening function is introduced in Equation 4.6 that can penalize longer paths based on their respective lengths. The quantity of dampening will depend on a parameter γ and the length of the path l.

$$\psi^l \leftarrow \gamma^{(l-2)} \tag{4.6}$$

4.2.5 Likelihood Score Computation for Higher Order Paths

The final likelihood score can be calculated iteratively by summing up the significance of higher order (longer) paths to their shorter counterparts. Every path of length greater than two can be seen as a shorter path by one hop connected to an edge. So the likelihood score based on a path would be a combination of the likelihood of a shorter path with the additional edge. This process is continued till the desired length of path l_{max} is achieved. Based on the three-degree phenomenon, the maximum l_{max} to be considered should be six based on the region of influence of each node to be three hops. However, in the result evaluation phase, experiments with l_{max} of length four are also performed, where it is assumed that the region of node influence is restricted to two hops instead of the usual three. The case of $l_{max} = 2$ is trivial as it is equivalent to standard common neighborhood-based algorithms. Through Equation 4.7, probabilities on G_{HOPLP} are calculated using weights from A_{HOPLP}. This equation represents an iterative procedure used to calculate likelihoods for increasingly longer paths starting from path-length 3. For path-length $l = 3$, $Prev_Iter_Impact$ would be likelihoods calculated using $l = 2$ paths (common neighbor-based similarity). The $Edge_Impact$ represents the effect of adding a new edge to an already existing path for likelihood estimation in longer paths. Finally, ψ is the dampening factor for penalizing and adequately representing information flow over a longer path. These calculations are mainly carried out in matrix form, as shown in Algorithm 2 in lines 1-5. A working example can be found in Section 4.2.6, which takes a small graph and calculates link likelihoods for longer paths. These are transformed (unpacked) into layer-specific probabilities using Equation 4.8, which uses parameters based on layer densities defined in Equation 4.3.

$$LI(n_1, n_2) \leftarrow \Bigg(\sum_{l=3}^{l_{max}} Prev\_Iter\_Impact * \\ Edge\_Impact * \psi^{l} \Bigg) \tag{4.7}$$

$$LI_j(n_1, n_2) \leftarrow LI(n_1, n_2) * DCZ(j) \tag{4.8}$$

Algorithm 2: HOPLP-MUL: Higher Order Path based Link Prediction in Multiplex Network

```
Input: Social Networks: $G_i(V, E_i), G_{HOPLP}$
Output: Likelihood Index: $LI$
                                        ▷ Summarization, Initialization
Create a summarized network $A_{HOPLP}$ from $n$ layered multiplex network on same
  user set using Equation 4.2.2
$prior_{\|V\|*\|V\|} \leftarrow 0$
$score_{\|V\|*\|V\|} \leftarrow 0$
                                        ▷ Initial Significance Computation
for each node pair $(n_1, n_2) \in G_{HOPLP}$ do
    ▷ Calculate $IS(n_1, n_2)$ using Equation 4.4 or Equation 4.5
    $score[n_1][n_2] \leftarrow IS(n_1, n_2)$
    $prior[n_1][n_2] \leftarrow score[n_1][n_2]$
                    ▷ Computation based on Higher Order Paths iteratively
for each Path Length $l > 2$ till $l_{max}$ do
    $curr_{\|V\|*\|V\|} \leftarrow 0$
    for each node pair $(n_1, n_2) \in G_{HOPLP}$ do
        for each Node neighbor $c \in N(n_1)$ do
            ▷ $Prev\_Iter\_Impact * Edge\_Impact * Penalty$ (Equation 4.7)
            $curr[n_1][n_2] = curr[n_1][n_2] + (score[n_1][c] * prior[c][n_2] * \psi^l)$
                                        ▷ Score update for longer path
    for each node pair $(n_1, n_2) \in G_{HOPLP}$ do
        $score[n_1][n_2] = score[n_1][n_2] + curr[n_1][n_2]$
        $prior[n_1][n_2] = curr[n_1][n_2]$
Return $LI$;
```

4.2.6 $HOPLP-MUL$ Algorithm with an illustrative example

Algorithm 2 demonstrates how the likelihood score matrix is calculated for graph G_{HOPLP}. The input to the algorithm is the summarized graph G_{HOPLP}, summarized weight matrix A_{HOPLP} and layer graphs $G_i \| i \in (1, n)$. The output is the likelihood matrix

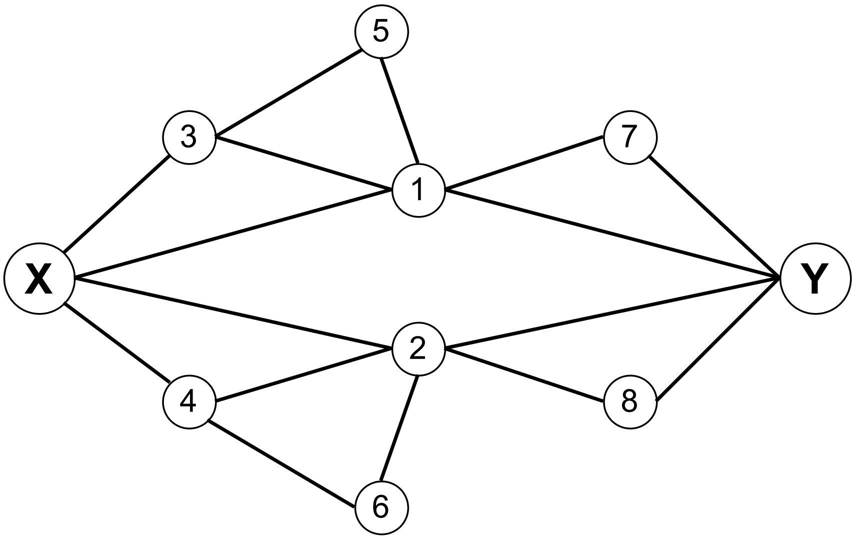


FIGURE 4.3: Example Graph for demonstrating working of $HOPLP-MUL$ for likelihood calculation over higher order paths.

of dimension $\|V\| * \|V\|$ such that $\|V\|$ is the number of nodes in G_{HOPLP}. The algorithm can be divided into three significant modules - initialization and graph summarization (lines 1-4), common neighborhood-based initial significance computations (lines 5-9), and a module which over successively larger path lengths to calculate cumulative significance based on higher-order paths (lines 10-20). The last iterative module can be divided into two sub-modules: current significance computation (lines 12-16) and score updation for combining this current score with the score from the previous iteration (lines 17-20).

For the sake of clarity and understanding purposes, an explanation of the working of the algorithm is provided using Fig.4.3. In this graph, we assume that the weight of all edges of the graph is 1, and the value of parameter $\gamma = 0.1$. The problem of finding link probability between nodes $X\&Y$ is considered. In Table 4.1, the last two rows and columns represent $X\&Y$ respectively in each matrix. In this table, it can be seen that the exact matrices where $PRIOR$ as in **Algorithm 2** represents $Edge\_Impact$ and $SCORE$ is $Prev\_Iter\_Impact$. The last two rows and columns represent $X\&Y$ respectively in each matrix. First, the effect of 2-length paths in graph are calculated as shown in Fig.4.3. Two such paths exist - $X-1-Y$ and $X-2-Y$. Using Equation 4.4, $IS(X,Y) = 0.4$ as

TABLE 4.1: SCORE and PRIOR matrices from Example for path-length $l = 2,3,4$.

PATH-LENGTH		PRIOR-MATRIX											SCORE-MATRIX										
	NODES	1	2	3	4	5	6	7	8	X	Y	NODES	1	2	3	4	5	6	7	8	X	Y
l = 2	1	0	0.5	0.75	0.25	0.33	0	0.25	0.25	0.33	0.5	1	0	0.5	0.75	0.25	0.33	0	0.25	0.25	0.33	0.5
	2	0.5	0	0.25	0.75	0	0.33	0.25	0.25	0.33	0.5	2	0.5	0	0.25	0.75	0	0.33	0.25	0.25	0.33	0.5
	3	0.75	0.25	0	0.25	0.2	0	0.2	0	0.2	0.2	3	0.75	0.25	0	0.25	0.2	0	0.2	0	0.2	0.2
	4	0.25	0.75	0.25	0	0	0.2	0	0.2	0.2	0.2	4	0.25	0.75	0.25	0	0	0.2	0	0.2	0.2	0.2
	5	0.33	0	0.2	0	0	0	0.2	0	0.53	0.2	5	0.33	0	0.2	0	0	0	0.2	0	0.53	0.2
	6	0	0.33	0	0.2	0	0	0	0.2	0.53	0.2	6	0	0.33	0	0.2	0	0	0	0.2	0.53	0.2
	7	0.25	0.25	0.2	0	0.2	0	0	0.25	0.2	0.2	7	0.25	0.25	0.2	0	0.2	0	0	0.25	0.2	0.2
	8	0.25	0.25	0	0.2	0	0.2	0.25	0	0.2	0.2	8	0.25	0.25	0	0.2	0	0.2	0.25	0	0.2	0.2
	X	0.33	0.33	0.2	0.2	0.53	0.53	0.2	0.2	0	0.4	X	0.33	0.33	0.2	0.2	0.53	0.53	0.2	0.2	0	0.4
	Y	0.5	0.5	0.2	0.2	0.2	0.2	0.2	0.2	0.4	0	Y	0.5	0.5	0.2	0.2	0.2	0.2	0.2	0.2	0.4	0
	NODES	1	2	3	4	5	6	7	8	X	Y	NODES	1	2	3	4	5	6	7	8	X	Y
l = 3	1	0	0.04	0.01	0.01	0.02	0	0.01	0.01	0.02	0.01	1	0	0.54	0.76	0.26	0.34	0	0.26	0.26	0.34	0.5
	2	0.04	0	0.01	0.01	0	0.02	0.01	0.01	0.02	0.01	2	0.54	0	0.26	0.76	0	0.34	0.26	0.26	0.34	0.5
	3	0.01	0.01	0	0	0.02	0	0.02	0	0.02	0.04	3	0.76	0.26	0	0.25	0.22	0	0.22	0	0.22	0.24
	4	0.01	0.01	0	0	0	0.02	0	0.02	0.02	0.04	4	0.26	0.76	0.25	0	0	0.22	0	0.22	0.22	0.24
	5	0.02	0	0.02	0	0	0	0.01	0	0.01	0.02	5	0.34	0	0.22	0	0	0	0.21	0	0.54	0.22
	6	0	0.02	0	0.02	0	0	0	0.01	0.01	0.02	6	0	0.34	0	0.22	0	0	0	0.21	0.54	0.22
	7	0.01	0.01	0.02	0	0.01	0	0	0	0.01	0.01	7	0.26	0.26	0.22	0	0.21	0	0	0.25	0.21	0.21
	8	0.01	0.01	0	0.02	0	0.01	0	0	0.01	0.01	8	0.26	0.26	0	0.22	0	0.21	0.25	0	0.21	0.21
	X	0.02	0.02	0.02	0.02	0.01	0.01	0.01	0.01	0	0.03	X	0.34	0.34	0.22	0.22	0.54	0.54	0.21	0.21	0	0.43
	Y	0.01	0.01	0.04	0.04	0.02	0.02	0.01	0.01	0.03	0	Y	0.5	0.5	0.24	0.24	0.22	0.22	0.21	0.21	0.43	0
	NODES	1	2	3	4	5	6	7	8	X	Y	NODES	1	2	3	4	5	6	7	8	X	Y
l = 4	1	0	0	0	0	0	0	0	0	0	0	1	0	0.54	0.76	0.26	0.34	0	0.26	0.26	0.34	0.5
	2	0	0	0	0	0	0	0	0	0	0	2	0.54	0	0.26	0.76	0	0.34	0.26	0.26	0.34	0.5
	3	0	0	0	0	0	0	0	0	0	0	3	0.76	0.26	0	0.25	0.22	0	0.22	0	0.22	0.24
	4	0	0	0	0	0	0	0	0	0	0	4	0.26	0.76	0.25	0	0	0.22	0	0.22	0.22	0.24
	5	0	0	0	0	0	0	0	0	0	0	5	0.34	0	0.22	0	0	0	0.21	0	0.54	0.22
	6	0	0	0	0	0	0	0	0	0	0	6	0	0.34	0	0.22	0	0	0	0.21	0.54	0.22
	7	0	0	0	0	0	0	0	0	0	0	7	0.26	0.26	0.22	0	0.21	0	0	0.25	0.21	0.21
	8	0	0	0	0	0	0	0	0	0	0	8	0.26	0.26	0	0.22	0	0.21	0.25	0	0.21	0.21
	X	0	0	0	0	0	0	0	0	0	0	X	0.34	0.34	0.22	0.22	0.54	0.54	0.21	0.21	0	0.43
	Y	0	0	0	0	0	0	0	0	0	0	Y	0.5	0.5	0.24	0.24	0.22	0.22	0.21	0.21	0.43	0

the contribution of nodes 1&2 is 0.2 each (5 edges each). Then 3 length paths between $X\&Y$ are considered, which are not calculated directly but use the values of the $SCORE$ matrix from the same iteration but to common neighbors instead of between $X\&Y$. The values of $SCORE(X,1) = SCORE(X,2) = 0.33$ and $SCORE(Y,1) = SCORE(Y,2) = 0.5$ are used. Hence the $CURR$ value is calculated for path $X-1-Y$ as $SCORE(X,1) * PRIOR(1,Y) * penalty = 0.015$ and similarly for path $X-2-Y$. Combining these with older path-length $l = 2$ $SCORE$, the total probability comes as 0.43. A point to be noted here is that in $HOPLP-MUL$ algorithm, we do not directly calculate the higher-order paths between $X\&Y$. For path-length $l = 4$, it can be observed that the $CURR$ matrix has become null after computation. This is because of the setup rounding off decimals to 2 significant places. Hence the $SCORE$ matrix for path-length $l = 3\&4$ remains the same. In this example, it is assumed that the node influence of nodes $X\&Y$ extends only a 2-hop region away from them ($l_{max} = 4$) for the sake of simplicity and keeping in mind the example size of the graph. This exact mechanism can be extended to a 3-hop region ($l_{max} = 6$), predicted by the 3 Degree of Influence phenomenon.

4.2.7 Complexity Analysis

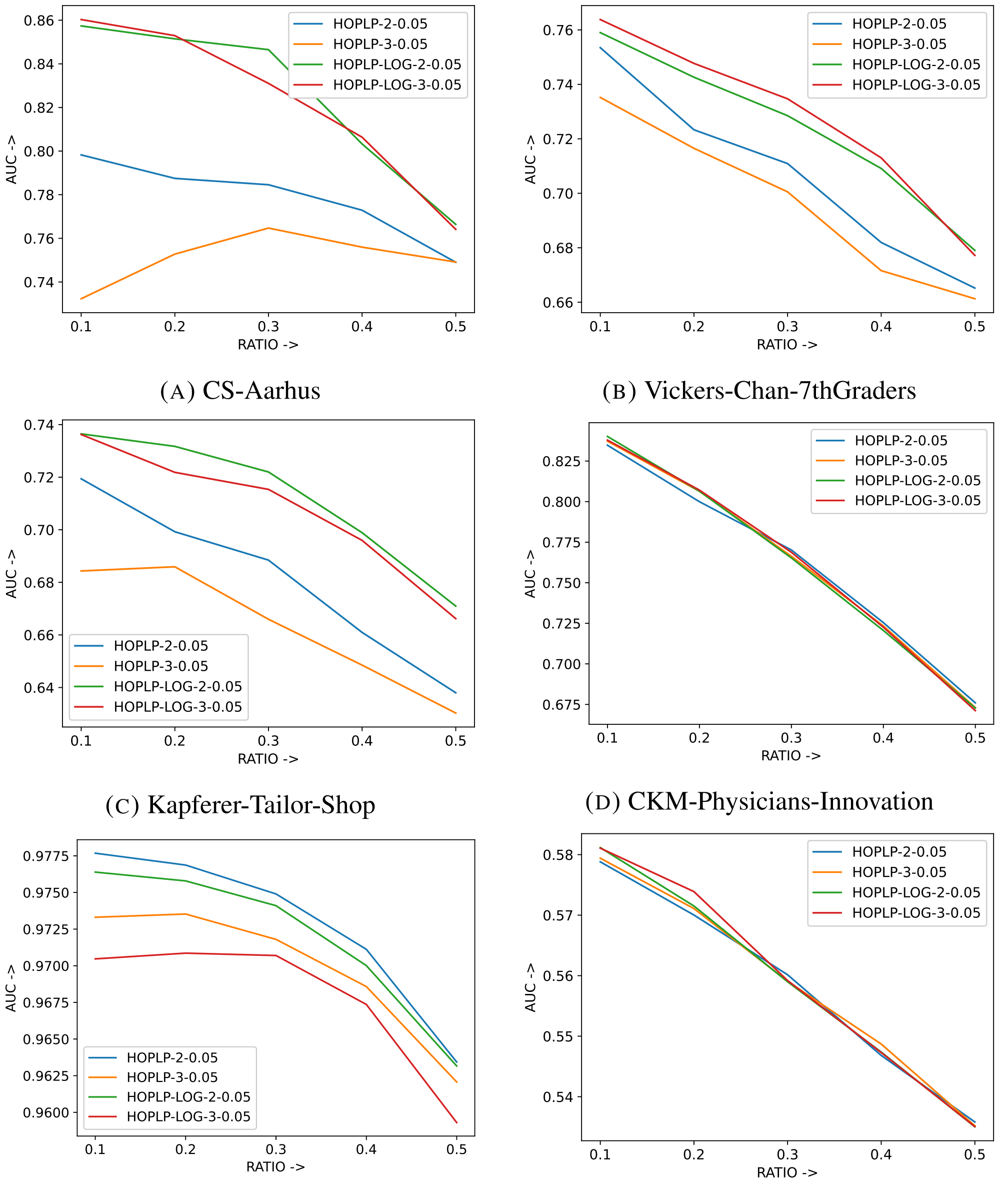
In this section, the complexity of the proposed algorithm $HOPLP-MUL$ is analyzed. Line 1 generates multiplex network in $\mathcal{O}(V+E)$ time. Line 2-3 initialize matrices which is a trivial process hence it takes $\mathcal{O}(1)$ time. The loop in line 4 for iteration has $\mathcal{O}(|V|^2)$ complexity. Equations 4.4,4.5 are used in line 5 and takes $\mathcal{O}(D_{avg})$ time. Lines 6 and 7 are trivial and hence take $\mathcal{O}(1)$ time. The module for initial significance calculation i.e., lines 4-7, collectively has $\mathcal{O}(|V|^2 * D_{avg})$ complexity. Line 8 iterates over the third module $l_{max} - 2$ times. Lines 10 iterates over all possible node pairs and hence has a total of $|V|^2$ iterations. The complexity of lines 11-13 has complexity of $\mathcal{O}(D_{avg})$ because of iteration over a set which contains neighbors of a node. Hence the total complexity of sub-module in lines 10-13 is $\mathcal{O}(|V|^2 * D_{avg})$. Similarly the complexity of sub-module which handles score updation is $\mathcal{O}(|V|^2)$. The total complexity of the third module in lines

8-16 is $\mathcal{O}((l_{max}-2)*|V|^2*D_{avg})$. Comparing the three major modules of $HOPLP-MUL$ algorithm, the combined complexity is $\mathcal{O}((V+E)+|V|^2*D_{avg}+(l_{max}-2)*|V|^2*D_{avg})$. Taking the most significant terms this can be simplified to $\mathcal{O}((l_{max}-1)*|V|^2*D_{avg})$. Since the $l_{max}-1$ part can be seen to much smaller than other terms, the complexity can be simplified to $\mathcal{O}(|V|^2*D_{avg})$.

4.3 Performance Analysis

4.3.1 Algorithm Variation Comparison

In this section, the performance of the proposed algorithm is compared with different baseline algorithms. The relationship between the algorithm's performance based on different forms of initial significance calculations (Equations 4.4 and 4.5) and different regions of influence of nodes ($l_{max}=4,6$) is also investigated. Hence the experiment are performed with the variations $(HOPLP-MUL-2-0.05, HOPLP-MUL-3-0.05$, $HOPLP-MUL-LOG-2-0.05$, $HOPLP-MUL-LOG-3-0.05$ such that LOG represents calculation from Equation 4.5 and $2,3$ represent $l_{max}=4,6$ respectively. Three metrics in these experiments: AUC, F1 Score, and Balanced Accuracy Score are used. Five different ratios (0.1, 0.2, 0.3, 0.4, 0.5) of testing set edges to total edges of graph datasets are considered. For the parameter γ, we experiment for values from $0.01-0.15$ with an interval of 0.02. Each of these tests is performed on six real-world networks. For the sake of simplicity in performing comparisons between different variations, the value of γ is fixed at 0.05 for Section 4.3.1. This setup helps streamline the results such that only differences caused due to changes of different initial significance and influence regions can be measured.



(A) CS-Aarhus

(B) Vickers-Chan-7thGraders

(C) Kapferer-Tailor-Shop

(D) CKM-Physicians-Innovation

(E) Pierreauger

(F) Rattus-Genetic

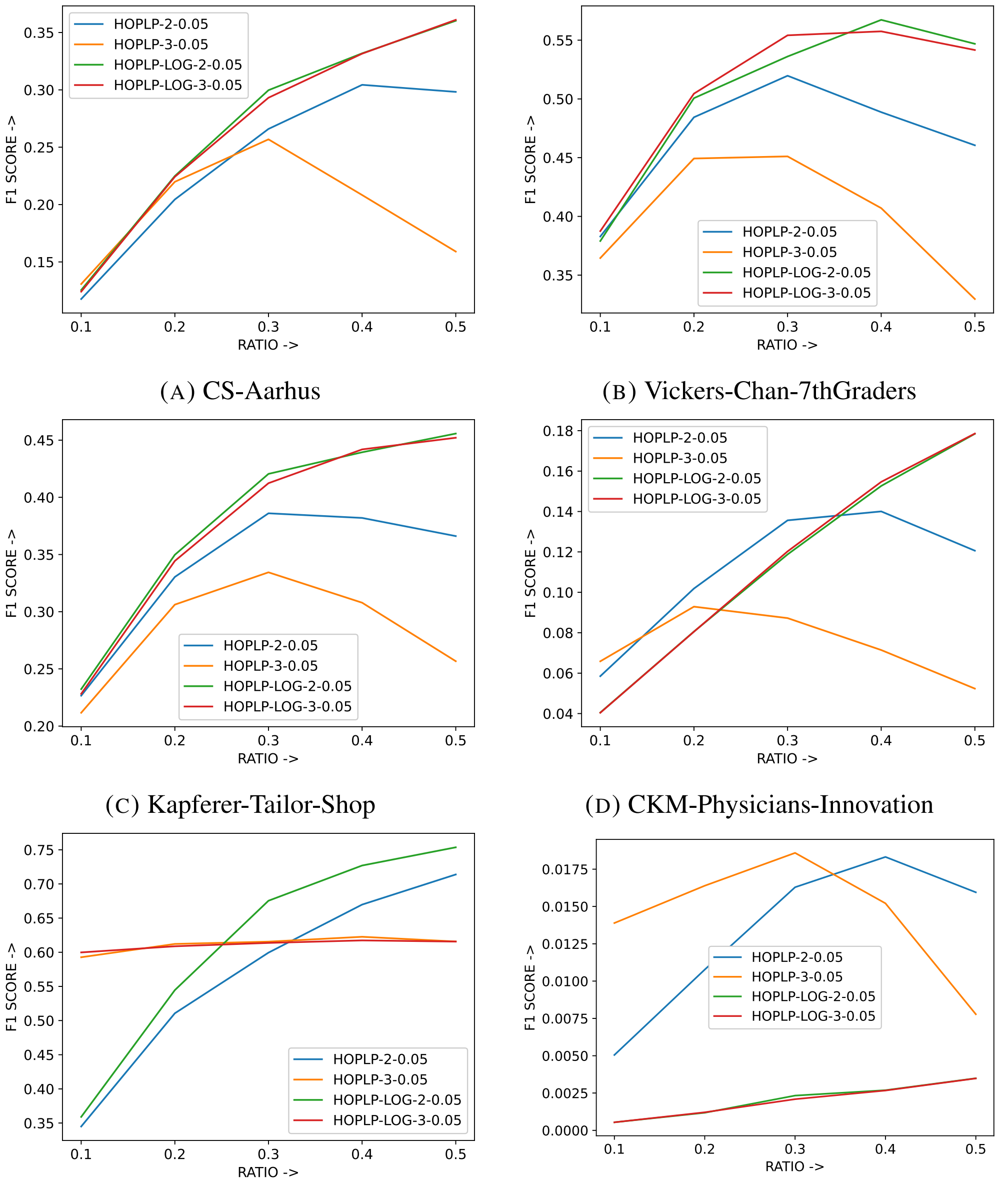
FIGURE 4.4: AUC comparison of $HOPLP - MUL$ variations for different feature sets on six datasets

4.3.1.1 Analysis of AUC Pattern on different algorithm variations

Fig. 4.4 presents the comparison of different algorithm variations on six datasets. In five datasets, it is observed that $HOPLP-MUL-LOG-2-0.05$ can be considered either the best performing algorithm, or it narrowly misses the best position. $HOPLP-MUL-LOG-3-0.05$ is the second-best performing algorithm in five datasets. The exception is Pierreauger, where it becomes the worst performing algorithm. $HOPLP-MUL-2-0.05$ can be considered the algorithm with the most middle-of-the-pack performance. It is tough to differentiate between the performance of algorithms in CKM-Physicians-Innovation and Rattus-Genetic datasets.

4.3.1.2 Analysis of F1 Score Pattern on different different variations

Fig. 4.5 presents the comparison of different algorithm variations on six datasets. $HOPLP-MUL-LOG-2-0.05$ can be observed as the best performing algorithm across all datasets except Rattus-Genetic. $HOPLP-MUL-LOG-3-0.05$ can be considered to have a similar performance to $HOPLP-MUL-LOG-2-0.05$ for all datasets except Pierreauger. This dataset is an exception because algorithms that consider larger influence regions of nodes, i.e., $HOPLP-MUL-3-0.05$ and $HOPLP-MUL-LOG-3-0.05$, show almost constant performance relative to $Ratio$, which for other algorithms and datasets has an increasing pattern. However, it can be observed that the quantum of increase decreases with the increase of the $Ratio$ variable. In other datasets, the performance of $HOPLP-MUL-3-0.05$ tends to fall after reaching $Ratio = 0.3$. The algorithms using exponential initial significance function, i.e., $HOPLP-MUL-2-0.05$ and $HOPLP-MUL-3-0.05$, show better performance in the Rattus-Genetic dataset contrary to the general pattern where logarithmic based variations tend to be better.



(A) CS-Aarhus

(B) Vickers-Chan-7thGraders

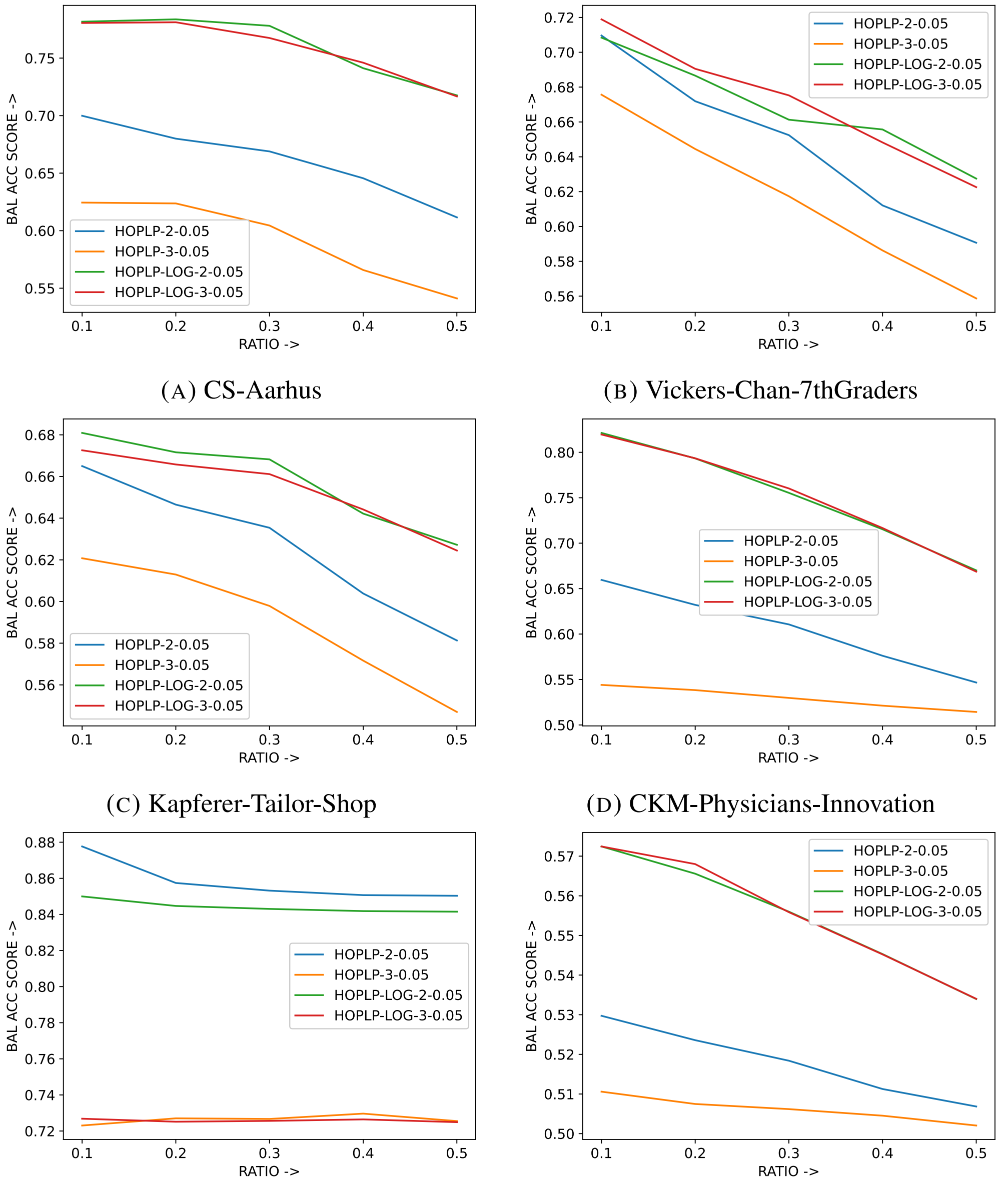
(C) Kapferer-Tailor-Shop

(D) CKM-Physicians-Innovation

(E) Pierreauger

(F) Rattus-Genetic

FIGURE 4.5: F1 Score comparison of $HOPLP-MUL$ variations for different feature sets on six datasets



(A) CS-Aarhus

(B) Vickers-Chan-7thGraders

(C) Kapferer-Tailor-Shop

(D) CKM-Physicians-Innovation

(E) Pierreauger

(F) Rattus-Genetic

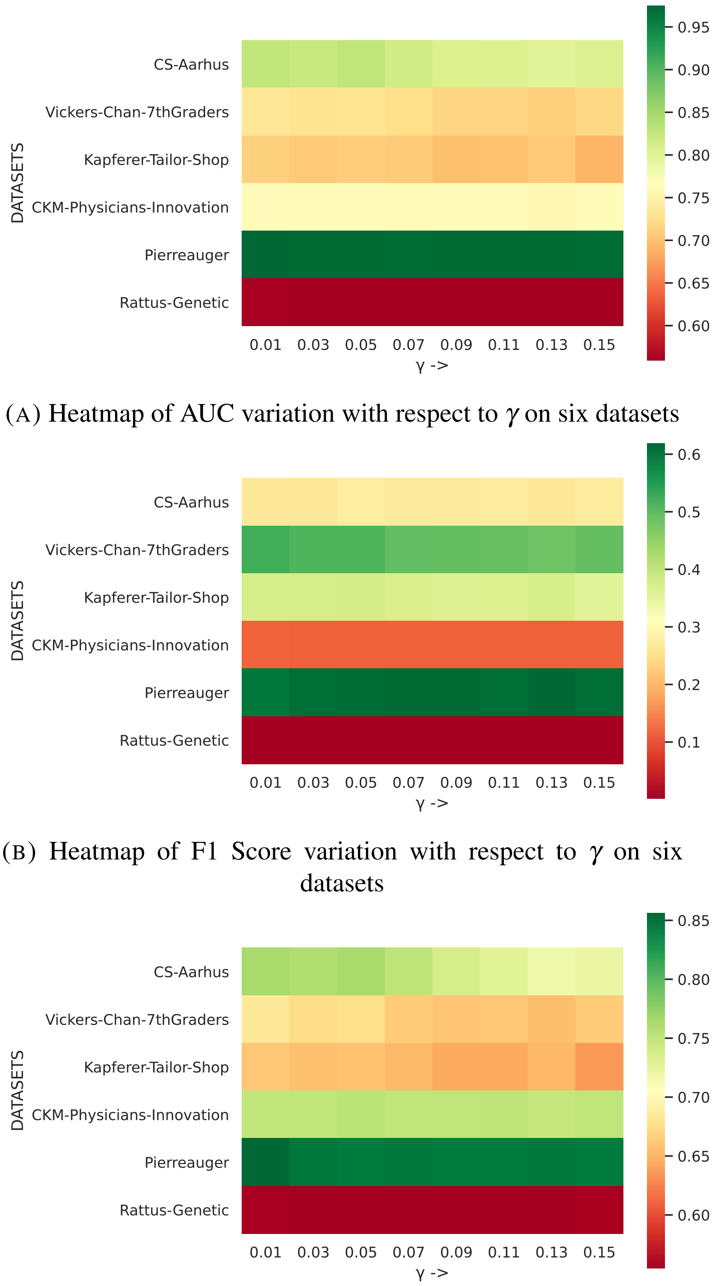
FIGURE 4.6: Balanced Accuracy Score comparison of $HOPLP-MUL$ variations for different feature sets on six datasets

4.3.1.3 Analysis of Balanced Accuracy Score Pattern on different algorithm variations

Fig. 4.6 presents the comparison of different algorithm variations on six datasets. In five datasets, it is observed that $HOPLP-MUL-LOG-2-0.05$ can be considered either the best performing algorithm or narrowly misses the best position. The only sizable difference is observed in the dataset Pierreauger where $HOPLP-MUL-2-0.05$ is the best performing algorithm. In this dataset, it is observed that the same performance by $HOPLP-MUL-3-0.05$ and $HOPLP-MUL-LOG-3-0.05$, which shows it is suitable for variations to consider larger regions of influence for prediction. However, $HOPLP-MUL-3-0.05$ is the worst-performing algorithm in all other datasets. $HOPLP-MUL-2-0.05$ shows a middle-of-the-pack performance in all datasets except Pierreauger, as explained above. $HOPLP-MUL-LOG-3-0.05$ shows excellent performance comparable to $HOPLP-MUL-LOG-2-0.05$ in all datasets except Pierreauger.

4.3.2 Variation of $HOPLP-MUL$ based on different γ values

Fig. 4.7 shows the comparison of the performance of $HOPLP-MUL$ algorithm for different values of γ. Based on variation comparisons in Section 4.3.1, $HOPLP-LOG-2$ is used as the base method for finding the best γ. In Fig. 4.7a, it is observed that in the overall pattern, the AUC values of $HOPLP-MUL$ algorithm decrease as the γ value is increased. In datasets CKM-Physicians-Innovations, Pierreauger, and Rattus-Genetic, there is only a minor improvement. So the most suitable candidates for best AUC performance is $\gamma = 0.01$. In Fig. 4.7b, it is observed that although lower values of γ show the best performance, the total decrease between consequent values is relatively slight. Even then, it is evident that the best candidate is $\gamma = 0.01$. In Fig. 4.7c, a similar pattern as the AUC metric is observed but with more contrast, one of decreasing balanced accuracy score values with the increase in γ. Only



(A) Heatmap of AUC variation with respect to γ on six datasets

(B) Heatmap of F1 Score variation with respect to γ on six datasets

(C) Heatmap of Accuracy Score variation with respect to γ on six datasets

FIGURE 4.7: Variation of algorithm performance of $HOPLP-MUL$ with respect to γ on six datasets

in the Rattus-Genetic dataset, the quantum of decrease is relatively tiny with increasing γ value. Hence, it can be concluded that the best performance of the algorithm can be achieved at $\gamma = 0.01$. Based on the results of Section 4.3.1 and 4.3.2, from hereon, the candidate variation $HOPLP - LOG - 2 - 0.01$ is referred to as the algorithm $HOPLP - MUL$ as the one with the best trade-off between accuracy and complexity.

4.3.3 $HOPLP - MUL$ comparison with link prediction methods on summarized weighted graph

The AUC metric is used in Table 4.2 to compare the proposed $HOPLP - MUL$ algorithm to baseline approaches. In five datasets, $HOPLP - MUL$ is the method with best results. In CS-Aarhus, CKM-Physicians-Innovation, and Pierreauger, the improvement is quite drastic. For Vickers-Chan-7thGraders and Kapferer-Tailor-Shop, the performance improvement is significant. In Rattus-Genetic, the proposed $HOPLP - MUL$ algorithm is second best behind CC-WT. The assessment of the suggested $HOPLP - MUL$ algorithm with respect to baseline approaches in terms of the F1 score is shown in Table 4.3. $HOPLP - MUL$ outperforms all other algorithms in four datasets. The exceptions are CS-Aarhus and Vickers-Chan-7thGraders datasets, in which for higher $Ratio$ values $(0.4, 0.5)$, the performance of $HOPLP - MUL$ algorithm becomes marginally less than most of the baseline methods. Table 4.4 compares the proposed $HOPLP - MUL$ algorithm to baseline approaches in terms of Balanced Accuracy score. $HOPLP - MUL$ outperforms all other algorithms in all six datasets. The most negligible improvement is seen for Kapferer-Tailor-Shop and Rattus-Genetic datasets, while all others show significant improvement.

TABLE 4.2: Comparison of the proposed algorithm $HOPLP-MUL$ with baseline algorithms in terms of AUC on six datasets and five *Ratio* values for testing to total edges percentage

DATASET	Ratio	CN-WT	JC-WT	PA-WT	AA-WT	RA-WT	CC-WT	LOCALP-WT	HOPLP-MUL
CS-Aarhus	0.1	0.64595	0.64335	0.54614	0.64816	0.65096	0.44883	0.61613	0.82107
	0.2	0.64674	0.6508	0.54807	0.64989	0.65166	0.48473	0.61786	0.81494
	0.3	0.64796	0.64695	0.54667	0.65017	0.64938	0.4763	0.62041	0.79835
	0.4	0.64107	0.64189	0.54389	0.63978	0.64296	0.47303	0.61972	0.78491
	0.5	0.62953	0.62924	0.5434	0.63019	0.63121	0.51155	0.61666	0.7437
Vickers-Chan-7thGraders	0.1	0.60084	0.5987	0.58839	0.60487	0.60348	0.58815	0.59864	0.72004
	0.2	0.59372	0.59351	0.58186	0.59796	0.59703	0.60575	0.59528	0.69263
	0.3	0.59094	0.58578	0.58154	0.59544	0.59457	0.58773	0.59104	0.69143
	0.4	0.59192	0.58385	0.57859	0.5926	0.59347	0.56523	0.58703	0.66945
	0.5	0.58715	0.57883	0.57524	0.59118	0.59074	0.53464	0.5822	0.62946
Kapferer-Tailor-Shop	0.1	0.58056	0.57322	0.56709	0.58032	0.58257	0.54942	0.57723	0.6887
	0.2	0.58108	0.57089	0.56704	0.57898	0.57952	0.53922	0.57612	0.69555
	0.3	0.57838	0.56688	0.56016	0.58046	0.57921	0.55687	0.57286	0.68425
	0.4	0.57611	0.5639	0.5607	0.57916	0.57823	0.52899	0.5661	0.67274
	0.5	0.57474	0.56289	0.5595	0.57438	0.57635	0.53624	0.56324	0.64621
CKM-Physicians-Innovation	0.1	0.66488	0.67271	0.52436	0.66903	0.66665	0.4514	0.67969	0.83184
	0.2	0.65093	0.65162	0.52151	0.6519	0.65312	0.46986	0.6735	0.7993
	0.3	0.63531	0.63339	0.5202	0.63347	0.63432	0.47539	0.65906	0.75949
	0.4	0.61306	0.61052	0.52	0.61185	0.61232	0.49387	0.64103	0.72289
	0.5	0.58772	0.58759	0.51919	0.58784	0.58817	0.49374	0.61609	0.67489
Pierreauger	0.1	0.73238	0.73283	0.59374	0.73289	0.7328	0.80174	0.71658	0.99041
	0.2	0.73289	0.73333	0.59606	0.73337	0.73336	0.80434	0.71929	0.98686
	0.3	0.73267	0.73278	0.5992	0.73353	0.73315	0.80005	0.72199	0.98197
	0.4	0.73151	0.73189	0.60187	0.7318	0.73172	0.79704	0.72296	0.9746
	0.5	0.72848	0.72838	0.60586	0.7292	0.72908	0.78689	0.72441	0.96476
Rattus-Genetic	0.1	0.53864	0.538	0.46466	0.53723	0.53961	0.63226	0.54274	0.58256
	0.2	0.53117	0.53234	0.48039	0.53251	0.53497	0.58736	0.54653	0.56995
	0.3	0.52758	0.52814	0.49553	0.52863	0.52859	0.58564	0.54478	0.5581
	0.4	0.52263	0.52338	0.5087	0.52229	0.52428	0.56224	0.54181	0.54526
	0.5	0.51704	0.51744	0.51869	0.51829	0.51729	0.56588	0.53589	0.53649

TABLE 4.3: Comparison of the proposed algorithm $HOPLP-MUL$ with baseline algorithms in terms of F1 Score on six datasets and five $Ratio$ values for testing to total edges percentage

DATASET	Ratio	CN-WT	JC-WT	PA-WT	AA-WT	RA-WT	CC-WT	LOCALP-WT	HOPLP-MUL
CS-Aarhus	0.1	0.07555	0.07062	0.05396	0.07545	0.07981	0.03379	0.07307	0.11176
	0.2	0.14319	0.13538	0.09811	0.13673	0.14704	0.07437	0.13791	0.19036
	0.3	0.19133	0.19271	0.13662	0.19468	0.19923	0.1026	0.19249	0.22585
	0.4	0.24614	0.24743	0.17019	0.24067	0.24788	0.12806	0.24697	0.24582
	0.5	0.28534	0.28563	0.1969	0.28244	0.28502	0.16978	0.28744	0.26939
Vickers-Chan-7thGraders	0.1	0.24653	0.22783	0.23589	0.25154	0.26361	0.25892	0.2412	0.32824
	0.2	0.36528	0.35041	0.34759	0.37409	0.38484	0.38864	0.36191	0.40257
	0.3	0.43054	0.41602	0.41382	0.44106	0.44419	0.42957	0.43104	0.44067
	0.4	0.46912	0.45963	0.45652	0.46569	0.47429	0.45965	0.47015	0.4287
	0.5	0.47812	0.47418	0.47275	0.48437	0.47959	0.46978	0.48498	0.39926
Kapferer-Tailor-Shop	0.1	0.14258	0.13289	0.13946	0.14232	0.1481	0.13279	0.13954	0.21385
	0.2	0.23389	0.22214	0.22988	0.23572	0.24117	0.2118	0.23299	0.32749
	0.3	0.30026	0.28525	0.28933	0.3017	0.30988	0.28708	0.29893	0.37428
	0.4	0.34405	0.32686	0.3336	0.34398	0.34802	0.3065	0.33972	0.37917
	0.5	0.36882	0.35599	0.36642	0.36625	0.37019	0.34694	0.36402	0.37317
CKM-Physicians-Innovation	0.1	0.03986	0.04132	0.0075	0.04051	0.04013	0.00515	0.0346	0.04237
	0.2	0.07771	0.07766	0.01476	0.0778	0.07837	0.01098	0.0655	0.08185
	0.3	0.11334	0.11211	0.02179	0.11171	0.11264	0.01658	0.09306	0.11769
	0.4	0.13878	0.13654	0.0285	0.13816	0.13895	0.02332	0.11803	0.15259
	0.5	0.15114	0.1515	0.03502	0.15044	0.15101	0.02787	0.13933	0.17742
Pierreauger	0.1	0.07684	0.07679	0.02269	0.07716	0.07714	0.0188	0.0706	0.12235
	0.2	0.15351	0.15315	0.04474	0.15407	0.15386	0.03717	0.14165	0.25072
	0.3	0.22907	0.22865	0.06523	0.22966	0.22906	0.05479	0.21454	0.37076
	0.4	0.3028	0.30372	0.0857	0.30452	0.30469	0.07137	0.28323	0.48386
	0.5	0.37477	0.37212	0.1043	0.37396	0.37233	0.08904	0.34728	0.57477
Rattus-Genetic	0.1	0.00056	0.00037	0.00076	0.00054	0.00056	0.0002	0.00048	0.01531
	0.2	0.00112	0.00078	0.00152	0.00116	0.00121	0.00046	0.00082	0.01692
	0.3	0.00185	0.0014	0.00244	0.00188	0.00191	0.00074	0.00136	0.0167
	0.4	0.00262	0.00211	0.00325	0.00262	0.00274	0.0011	0.00225	0.01265
	0.5	0.00349	0.00283	0.00385	0.00359	0.00342	0.00167	0.00335	0.0139

TABLE 4.4: Comparison of the proposed algorithm $HOPLP-MUL$ with baseline algorithms in terms of Balanced Accuracy Score on six datasets and five *Ratio* values for testing to total edges percentage

DATASET	Ratio	CN-WT	JC-WT	PA-WT	AA-WT	RA-WT	CC-WT	LOCALP-WT	HOPLP-MUL
CS-Aarhus	0.1	0.60606	0.5953	0.54222	0.60664	0.6143	0.44283	0.60167	0.61016
	0.2	0.61497	0.60817	0.54215	0.60866	0.6204	0.47741	0.61045	0.6215
	0.3	0.61325	0.61406	0.54376	0.61547	0.61856	0.47607	0.61466	0.60273
	0.4	0.61844	0.61922	0.54454	0.61378	0.61897	0.47427	0.62047	0.59219
	0.5	0.61565	0.61588	0.54382	0.61391	0.61544	0.50167	0.6169	0.59369
Vickers-Chan-7thGraders	0.1	0.56258	0.53822	0.5504	0.56843	0.58162	0.56601	0.55599	0.64781
	0.2	0.57132	0.55183	0.55531	0.57972	0.59081	0.58289	0.56651	0.6093
	0.3	0.57865	0.55628	0.56392	0.58949	0.59374	0.55854	0.57691	0.59919
	0.4	0.58329	0.56508	0.57073	0.58305	0.59265	0.54492	0.58161	0.57838
	0.5	0.57801	0.56188	0.56922	0.58611	0.58761	0.52387	0.58235	0.54617
Kapferer-Tailor-Shop	0.1	0.5521	0.53511	0.5468	0.55165	0.56102	0.53478	0.54714	0.63293
	0.2	0.55611	0.54078	0.55168	0.55858	0.56475	0.52418	0.55499	0.63702
	0.3	0.56309	0.54435	0.55254	0.56424	0.57289	0.54035	0.56077	0.62428
	0.4	0.56636	0.54459	0.55481	0.56641	0.57148	0.51484	0.56139	0.60293
	0.5	0.56368	0.54748	0.55781	0.56286	0.56925	0.52088	0.55821	0.5858
CKM-Physicians-Innovation	0.1	0.66069	0.66849	0.5261	0.66436	0.66214	0.45726	0.66562	0.79291
	0.2	0.64789	0.64883	0.52586	0.64855	0.65006	0.4711	0.65936	0.76069
	0.3	0.6333	0.63184	0.5242	0.63133	0.63231	0.47528	0.64946	0.72851
	0.4	0.61199	0.60982	0.523	0.61066	0.61113	0.48834	0.63784	0.70231
	0.5	0.58717	0.58733	0.52181	0.58719	0.58754	0.48408	0.61435	0.67011
Pierreauger	0.1	0.71863	0.71856	0.62186	0.71855	0.71842	0.6817	0.71584	0.9576
	0.2	0.72121	0.72111	0.62396	0.7211	0.72103	0.68508	0.71847	0.95973
	0.3	0.723	0.72265	0.62486	0.72323	0.72282	0.68502	0.72076	0.96033
	0.4	0.72379	0.72385	0.62646	0.72362	0.72354	0.68381	0.72157	0.95813
	0.5	0.72271	0.72232	0.62737	0.72294	0.72275	0.6849	0.72128	0.95193
Rattus-Genetic	0.1	0.53745	0.51648	0.5701	0.53528	0.53774	0.49428	0.55026	0.51046
	0.2	0.53039	0.51503	0.5673	0.53124	0.53365	0.50655	0.54247	0.50833
	0.3	0.52706	0.51685	0.5661	0.5278	0.52777	0.50896	0.54212	0.50559
	0.4	0.52233	0.51597	0.56171	0.52182	0.52378	0.51199	0.54052	0.50388
	0.5	0.51691	0.51259	0.55713	0.51804	0.51704	0.51562	0.53538	0.50432

4.3.4 $HOPLP-MUL$ comparison with multiplex link prediction methods on individual layers

The results of $HOPLP-MUL$ algorithm's application to particular layers of multiplex networks are described in this section. The AUC measure is used in Tables 4.5 and 4.6, to compare the new $HOPLP-MUL$ algorithm with baseline approaches. Additionally, the technique is compared to two algorithms that are expressly intended for link prediction in multiplex networks, namely $NSILR-MUL$ and $MADM-MUL$, in this section. $HOPLP-MUL$ outperforms all other algorithms in all three datasets for $Ratio$ values of $0.1-0.3$, but especially for $Ratio = 0.4$. On layer-1 of the Kapferer-Tailor-Shop and CKM-Physicians-Innovation datasets, $NSILR-MUL$ and $MADM-MUL$ outperform $HOPLP-MUL$ approach. For conventional weighted link prediction techniques, the difference in AUC between the approach and the other benchmark algorithms lessens as the Ratio value increases. For $NSILR-MUL$ and $MADM-MUL$, the pattern of change as the $Ratio$ value increases is the inverse of the pattern for other algorithms, which results in improved performance on fewer edges for any given layer. $HOPLP-MUL$ algorithm performs better than $MVERSE-EMB$ for all layers of Vickers-Chan-7thGraders and layers 2 and 4 of Kapferer-Tailor-Shop dataset. For CKM-Physicians-Innovation dataset $HOPLP-MUL$ algorithm's performance is below $MVERSE-EMB$.

In Tables 4.7 and 4.8, the suggested $HOPLP-MUL$ algorithm is compared to standard approaches in terms of the F1 score. When compared to traditional weighted link prediction algorithms, $HOPLP-MUL$ outperforms them in all three datasets across all levels, except in the case of CKM-Physicians-Innovation dataset, where it becomes the fourth-best performing algorithm for lower $Ratio$ values $(0.1, 0.2)$ behind $CN-WT$, $AA-WT$, and $RA-WT$. But for higher $Ratio$ values $(0.3, 0.4, 0.5)$ it is the best performing algorithm for CKM-Physicians-Innovation dataset. In contrast to $NSILR-MUL$ and $MADM-MUL$, the approach outperforms both benchmarks for all $Ratio$ values between $0.1\&0.4$. In contrast to the AUC trend, all three $NSILR-MUL$,

TABLE 4.5: Comparison of the proposed algorithm $HOPLP-MUL$ with baseline algorithms in terms of AUC layer-wise on three datasets and five *Ratio* values for testing to total edges percentage

DATASET	Layer No.	Ratio	CN-WT	PA-WT	JC-WT	AA-WT	RA-WT	CC-WT	LOCALP-WT	NSILR-MUL	MADM-MUL	MVERSE-EMB	HOPLP-MUL
Vickers-Chan-7thGraders	1	0.1	0.59264	0.58011	0.59237	0.59553	0.59814	0.57576	0.5941	0.67383	0.64744	0.58373	0.77557
		0.2	0.59097	0.57682	0.58659	0.59292	0.59469	0.57026	0.58964	0.68865	0.67858	0.63411	0.75416
		0.3	0.58722	0.57641	0.58255	0.58999	0.59182	0.54356	0.58549	0.67236	0.66006	0.63743	0.72856
		0.4	0.58726	0.57378	0.57803	0.59082	0.59233	0.53047	0.58239	0.69457	0.67443	0.66327	0.70315
		0.5	0.58526	0.57038	0.57285	0.58728	0.58691	0.53176	0.57822	0.85148	0.86447	0.72753	0.66052
	2	0.1	0.61902	0.56741	0.61558	0.6267	0.62889	0.48223	0.59643	0.67613	0.76145	0.51797	0.83789
		0.2	0.62247	0.56606	0.61424	0.62223	0.63047	0.49103	0.59398	0.70182	0.77866	0.53423	0.80438
		0.3	0.61905	0.56638	0.60894	0.62182	0.626	0.48118	0.59322	0.7427	0.77261	0.628	0.78156
		0.4	0.61329	0.56402	0.61106	0.61451	0.62156	0.50157	0.59178	0.75734	0.79776	0.66767	0.75251
		0.5	0.60488	0.56501	0.59501	0.60629	0.60729	0.50581	0.58739	0.82937	0.87328	0.7149	0.6941
	3	0.1	0.62164	0.58049	0.6175	0.62069	0.62662	0.57283	0.60573	0.5735	0.5131	0.72242	0.81855
		0.2	0.62393	0.57656	0.61901	0.62257	0.62611	0.5692	0.60084	0.61915	0.54899	0.68271	0.80812
		0.3	0.62152	0.57468	0.61524	0.62353	0.62386	0.5556	0.60198	0.65218	0.63327	0.63888	0.78363
		0.4	0.62037	0.57429	0.61428	0.62196	0.62074	0.54368	0.6	0.6705	0.66265	0.625	0.7454
		0.5	0.61308	0.57382	0.6057	0.6086	0.61482	0.53727	0.60014	0.76442	0.75108	0.74885	0.71265
Kapferer-Tailor-Shop	1	0.1	0.60486	0.56184	0.59865	0.61264	0.60498	0.52083	0.58524	0.76062	0.80789	0.74507	0.7527
		0.2	0.60657	0.56166	0.59469	0.60498	0.60136	0.5045	0.58684	0.66902	0.75498	0.74991	0.72347
		0.3	0.59718	0.5625	0.59257	0.59885	0.59925	0.5177	0.58597	0.76188	0.7278	0.74108	0.70505
		0.4	0.59247	0.56256	0.58216	0.59263	0.58955	0.52191	0.58082	0.73548	0.74697	0.72225	0.68439
		0.5	0.58225	0.56422	0.57265	0.57901	0.57752	0.53199	0.57756	0.77726	0.81609	0.78406	0.64765
	2	0.1	0.5933	0.55929	0.5917	0.59403	0.59351	0.58928	0.57759	0.64462	0.583	0.55987	0.75132
		0.2	0.59517	0.55953	0.59036	0.5934	0.59279	0.58149	0.5755	0.63217	0.60123	0.64178	0.72748
		0.3	0.59318	0.55929	0.58669	0.59445	0.59601	0.5807	0.57592	0.62654	0.66526	0.69068	0.71071
		0.4	0.58828	0.55764	0.58892	0.59235	0.59297	0.5676	0.57596	0.64257	0.68733	0.66688	0.68361
		0.5	0.5825	0.55684	0.57512	0.58282	0.5831	0.5494	0.57362	0.74455	0.74659	0.7469	0.65287

TABLE 4.6: Comparison of the proposed algorithm $HOPLP-MUL$ with baseline algorithms in terms of AUC layer-wise on three datasets and five $Ratio$ values for testing to total edges percentage (contd..)

DATASET	Layer No.	Ratio	CN-WT	PA-WT	JC-WT	AA-WT	RA-WT	CC-WT	LOCALP-WT	NSILR-MUL	MADM-MUL	MVERSE-EMB	HOPLP-MUL
Kapferer-Tailor-Shop	3	0.1	0.59465	0.58391	0.59048	0.59799	0.60194	0.56674	0.58746	0.60643	0.78729	0.78484	0.69189
		0.2	0.59042	0.57579	0.58732	0.59641	0.58661	0.57093	0.58073	0.71011	0.69995	0.6448	0.66142
		0.3	0.58192	0.5776	0.5823	0.58149	0.58361	0.56323	0.58179	0.65467	0.66214	0.66339	0.64079
		0.4	0.56753	0.56995	0.57142	0.56861	0.56588	0.56265	0.57163	0.65029	0.6414	0.68635	0.61517
		0.5	0.55526	0.57049	0.54907	0.55341	0.55506	0.55853	0.55777	0.71653	0.69904	0.67651	0.58241
	4	0.1	0.606	0.57718	0.59673	0.60136	0.60341	0.5722	0.59919	0.58657	0.51445	0.60686	0.71608
		0.2	0.60078	0.57634	0.59186	0.59576	0.59168	0.55838	0.60304	0.54893	0.59805	0.62554	0.68153
		0.3	0.58098	0.57549	0.57476	0.58529	0.58001	0.57257	0.59071	0.57645	0.62804	0.57372	0.64587
		0.4	0.56782	0.57322	0.56679	0.5688	0.56651	0.56526	0.58244	0.52468	0.61332	0.63256	0.62955
		0.5	0.55495	0.57306	0.55111	0.558	0.55834	0.54834	0.57537	0.62987	0.65278	0.66367	0.5882
CKM-Physicians-Innovation	1	0.1	0.61102	0.55727	0.60968	0.608	0.61276	0.48	0.64181	0.83846	0.8049	0.9055	0.71401
		0.2	0.59254	0.55546	0.58915	0.59428	0.59421	0.48811	0.62607	0.82258	0.81245	0.85573	0.67139
		0.3	0.57691	0.55152	0.57343	0.5777	0.57623	0.5015	0.60527	0.83026	0.8066	0.87624	0.64145
		0.4	0.55793	0.54855	0.55706	0.55667	0.55735	0.50274	0.58433	0.8176	0.80742	0.88396	0.61027
		0.5	0.54326	0.54238	0.54133	0.54224	0.54195	0.50455	0.56289	0.81672	0.81927	0.89373	0.57975
	2	0.1	0.62515	0.52389	0.62195	0.62257	0.62672	0.43557	0.6508	0.69274	0.67307	0.8324	0.73769
		0.2	0.60593	0.52375	0.60393	0.60433	0.60364	0.45342	0.62561	0.67828	0.70683	0.82135	0.70117
		0.3	0.58412	0.52232	0.5841	0.58314	0.5837	0.46998	0.60867	0.6835	0.70655	0.84521	0.65952
		0.4	0.56375	0.52066	0.56396	0.56388	0.56441	0.48797	0.58479	0.68295	0.71381	0.86156	0.62149
		0.5	0.54599	0.51634	0.54695	0.54776	0.5481	0.48942	0.56361	0.72359	0.74841	0.88259	0.58745
	3	0.1	0.60595	0.51608	0.60165	0.59935	0.59927	0.47439	0.6197	0.50693	0.58319	0.67667	0.69687
		0.2	0.58671	0.5171	0.58584	0.58576	0.5842	0.48683	0.60487	0.50875	0.58819	0.69522	0.6615
		0.3	0.56984	0.51768	0.5694	0.57134	0.57012	0.4896	0.58872	0.5308	0.56816	0.71095	0.63111
		0.4	0.55439	0.51475	0.55331	0.55466	0.55339	0.49276	0.56967	0.54651	0.58292	0.70173	0.60377
		0.5	0.53836	0.51305	0.53938	0.53894	0.53906	0.49403	0.55117	0.5848	0.62415	0.75702	0.57415

TABLE 4.7: Comparison of the proposed algorithm $HOPLP-MUL$ with baseline algorithms in terms of F1 Score layer-wise on three datasets and five *Ratio* values for testing to total edges percentage

DATASET	Layer No.	Ratio	CN-WT	PA-WT	JC-WT	AA-WT	RA-WT	CC-WT	LOCALP-WT	NSILR-MUL	MADM-MUL	MVERSE-EMB	HOPLP-MUL
Vickers-Chan-7thGraders	1	0.1	0.23234	0.2198	0.21341	0.23703	0.24751	0.23074	0.22711	0.03538	0.03269	0.02817	0.36884
		0.2	0.35137	0.33263	0.33206	0.35722	0.36902	0.343	0.34857	0.08204	0.08616	0.07281	0.48971
		0.3	0.41243	0.39766	0.39899	0.41899	0.42884	0.38755	0.41402	0.16237	0.16881	0.15406	0.53922
		0.4	0.4486	0.44206	0.44207	0.45384	0.46286	0.41806	0.45155	0.32651	0.32604	0.30713	0.54678
		0.5	0.46538	0.45956	0.4552	0.46777	0.46607	0.45903	0.47055	0.73093	0.74251	0.55646	0.52696
	2	0.1	0.10942	0.09536	0.10508	0.11881	0.1306	0.07326	0.10219	0.01307	0.02225	0.00963	0.191
		0.2	0.18623	0.1666	0.18549	0.19162	0.22908	0.13247	0.18591	0.03656	0.05672	0.02215	0.31085
		0.3	0.24901	0.21787	0.24611	0.24902	0.27958	0.17645	0.25216	0.10315	0.1365	0.08404	0.37908
		0.4	0.30354	0.26385	0.30826	0.29892	0.30889	0.21908	0.30523	0.22518	0.29543	0.19163	0.42471
		0.5	0.33242	0.30017	0.33042	0.33154	0.33195	0.24781	0.32988	0.50117	0.65207	0.40939	0.42792
	3	0.1	0.16091	0.13369	0.14028	0.15392	0.15346	0.12536	0.14546	0.02887	0.02294	0.02897	0.23898
		0.2	0.24395	0.21794	0.22878	0.25072	0.25272	0.20568	0.23238	0.07093	0.0517	0.06226	0.37714
		0.3	0.29675	0.27762	0.29286	0.29755	0.31927	0.25776	0.29526	0.15756	0.10472	0.09635	0.44528
		0.4	0.34714	0.32164	0.34699	0.34529	0.34926	0.29383	0.34572	0.24787	0.23321	0.1882	0.46571
		0.5	0.37711	0.3533	0.37783	0.36839	0.37606	0.31429	0.37774	0.52324	0.44641	0.49561	0.46831
Kapferer-Tailor-Shop	1	0.1	0.07427	0.07089	0.06992	0.07494	0.07898	0.04998	0.07161	0.02082	0.01848	0.0235	0.10543
		0.2	0.13276	0.12813	0.13118	0.13418	0.13811	0.09055	0.13085	0.03534	0.04003	0.05272	0.17594
		0.3	0.1821	0.17427	0.18098	0.18125	0.18565	0.1264	0.18262	0.09088	0.078	0.0929	0.23802
		0.4	0.22294	0.21052	0.21849	0.22174	0.2205	0.15428	0.22031	0.16412	0.16132	0.17379	0.27766
		0.5	0.24558	0.23636	0.24025	0.23927	0.23913	0.18857	0.23836	0.35341	0.37883	0.41191	0.2931
	2	0.1	0.10429	0.09707	0.09689	0.10982	0.11103	0.09363	0.10329	0.02261	0.01838	0.01649	0.15025
		0.2	0.18211	0.16857	0.16894	0.18215	0.18897	0.16093	0.17428	0.0395	0.04457	0.0476	0.24435
		0.3	0.24112	0.21989	0.22364	0.23603	0.24636	0.21696	0.23556	0.08119	0.11703	0.10271	0.30916
		0.4	0.27473	0.26253	0.27378	0.27883	0.28436	0.25129	0.27277	0.16636	0.22874	0.1947	0.33968
		0.5	0.29691	0.29252	0.29456	0.29761	0.30121	0.27141	0.29675	0.43194	0.45028	0.44684	0.35898

TABLE 4.8: Comparison of the proposed algorithm $HOPLP-MUL$ with baseline algorithms in terms of F1 Score layer-wise on three datasets and five *Ratio* values for testing to total edges percentage (contd..)

DATASET	Layer No.	Ratio	CN-WT	PA-WT	JC-WT	AA-WT	RA-WT	CC-WT	LOCALP-WT	NSILR-MUL	MADM-MUL	MVERSE-EMB	HOPLP-MUL
Kapferer-Tailor-Shop	3	0.1	0.04704	0.03482	0.04729	0.04797	0.04912	0.02818	0.04673	0.00743	0.02184	0.01876	0.05635
		0.2	0.09182	0.06439	0.09283	0.09525	0.08897	0.05295	0.08622	0.01833	0.02508	0.01865	0.10425
		0.3	0.12372	0.08904	0.12875	0.12325	0.12518	0.0725	0.11969	0.03298	0.04533	0.04057	0.13309
		0.4	0.14541	0.11239	0.15598	0.14706	0.14531	0.09835	0.13714	0.06615	0.08715	0.09287	0.16635
		0.5	0.15248	0.13211	0.14472	0.14988	0.15243	0.11997	0.14278	0.20833	0.22034	0.19464	0.16313
	4	0.1	0.0539	0.04248	0.05299	0.05274	0.05737	0.03234	0.05298	0.01487	0.00606	0.01252	0.0618
		0.2	0.09755	0.07811	0.0977	0.09455	0.09648	0.05925	0.09932	0.02495	0.03786	0.02324	0.1079
		0.3	0.13081	0.11299	0.13029	0.13382	0.1308	0.08876	0.13106	0.05896	0.0814	0.04743	0.15024
		0.4	0.15224	0.13955	0.1548	0.15124	0.15027	0.10911	0.15578	0.06382	0.11305	0.09102	0.18598
		0.5	0.16357	0.16545	0.16018	0.16728	0.16809	0.12896	0.17322	0.20017	0.20173	0.20785	0.18873
CKM-Physicians-Innovation	1	0.1	0.03032	0.00422	0.03018	0.02945	0.03054	0.00245	0.02253	0.00469	0.01348	0.01398	0.03029
		0.2	0.05594	0.00836	0.05369	0.05656	0.0566	0.005	0.04459	0.01061	0.03222	0.02474	0.05568
		0.3	0.07733	0.01228	0.07431	0.0774	0.07641	0.00765	0.06515	0.02313	0.06463	0.05142	0.07932
		0.4	0.08621	0.01635	0.08542	0.08468	0.08597	0.0105	0.08118	0.04804	0.12515	0.11143	0.09732
		0.5	0.08888	0.01989	0.08535	0.08716	0.08667	0.0135	0.09094	0.11108	0.23608	0.24281	0.10422
	2	0.1	0.03712	0.00375	0.03641	0.03645	0.03754	0.00259	0.02748	0.00617	0.01295	0.01123	0.03657
		0.2	0.06896	0.0078	0.06801	0.06791	0.06783	0.00526	0.051	0.01319	0.03482	0.02085	0.07004
		0.3	0.09024	0.0115	0.0902	0.08921	0.09026	0.00794	0.07458	0.02801	0.06779	0.05484	0.09606
		0.4	0.10079	0.01536	0.10104	0.10124	0.10208	0.01108	0.09025	0.05751	0.12701	0.11186	0.11271
		0.5	0.09887	0.01875	0.10029	0.10193	0.10231	0.01398	0.09755	0.13954	0.23845	0.24409	0.11827
	3	0.1	0.04092	0.00329	0.03949	0.03878	0.03858	0.00262	0.0297	0.0023	0.01399	0.00316	0.03663
		0.2	0.0708	0.00647	0.07033	0.07007	0.06925	0.0055	0.0566	0.00602	0.03415	0.00863	0.06672
		0.3	0.09088	0.0096	0.09095	0.09304	0.09161	0.00822	0.07999	0.03062	0.04758	0.0228	0.09124
		0.4	0.10077	0.01237	0.09908	0.10111	0.09921	0.01071	0.09435	0.06182	0.08763	0.06097	0.10915
		0.5	0.09291	0.01538	0.09506	0.09399	0.09442	0.01289	0.0982	0.12961	0.17138	0.16394	0.11126

TABLE 4.9: Comparison of the proposed algorithm $HOPLP-MUL$ with baseline algorithms in terms of Balanced Accuracy Score layer-wise on three datasets and five *Ratio* values for testing to total edges percentage

DATASET	Layer No.	Ratio	CN-WT	PA-WT	JC-WT	AA-WT	RA-WT	CC-WT	LOCALP-WT	NSILR-MUL	MADM-MUL	MVERSE-EMB	HOPLP-MUL
Vickers-Chan-7thGraders	1	0.1	0.56342	0.54828	0.53813	0.56907	0.58082	0.56175	0.55698	0.64623	0.63125	0.58326	0.73055
		0.2	0.57266	0.55531	0.55013	0.57866	0.59062	0.55696	0.56949	0.63429	0.65912	0.60231	0.69728
		0.3	0.57415	0.56055	0.55433	0.58016	0.59138	0.53253	0.57322	0.62218	0.63996	0.61113	0.67372
		0.4	0.57837	0.56688	0.55992	0.58257	0.59203	0.52167	0.5786	0.64011	0.63803	0.62448	0.64739
		0.5	0.57638	0.56346	0.55444	0.58092	0.58236	0.52659	0.57833	0.77505	0.78723	0.6694	0.61304
	2	0.1	0.57881	0.54886	0.57224	0.59482	0.61177	0.48568	0.56612	0.55599	0.73898	0.49613	0.76793
		0.2	0.57762	0.55054	0.57643	0.58365	0.61821	0.48775	0.57738	0.58979	0.75469	0.48254	0.74881
		0.3	0.58547	0.55241	0.58259	0.58572	0.61037	0.48891	0.58883	0.65324	0.77113	0.60237	0.72805
		0.4	0.59324	0.5569	0.59702	0.58937	0.59743	0.49477	0.59452	0.69162	0.79962	0.63135	0.70329
		0.5	0.59323	0.56227	0.59146	0.5928	0.59287	0.49649	0.59123	0.74853	0.85768	0.66196	0.6672
	3	0.1	0.60595	0.56948	0.58134	0.59791	0.59845	0.56188	0.58746	0.58113	0.51705	0.63245	0.75949
		0.2	0.5984	0.57069	0.58359	0.60389	0.60532	0.55592	0.58666	0.59704	0.54868	0.62466	0.75459
		0.3	0.58947	0.57144	0.58588	0.5902	0.60796	0.54277	0.58813	0.64176	0.56173	0.57602	0.7279
		0.4	0.59494	0.57312	0.59474	0.5935	0.5971	0.53255	0.59372	0.64304	0.60044	0.57646	0.69259
		0.5	0.59621	0.57344	0.59702	0.59039	0.59629	0.51901	0.59698	0.71139	0.64658	0.68765	0.66205
Kapferer-Tailor-Shop	1	0.1	0.58236	0.57553	0.57355	0.58575	0.59023	0.50667	0.57766	0.73472	0.74219	0.67391	0.69667
		0.2	0.58235	0.57619	0.57918	0.58307	0.585	0.5036	0.57887	0.63586	0.71668	0.68402	0.67074
		0.3	0.58075	0.57551	0.57974	0.57988	0.58327	0.50212	0.58127	0.71306	0.69299	0.66321	0.66198
		0.4	0.58069	0.57333	0.57676	0.57954	0.57812	0.50073	0.57857	0.68368	0.70034	0.67011	0.64472
		0.5	0.57514	0.5699	0.57148	0.57077	0.57059	0.5142	0.57017	0.70742	0.74115	0.73242	0.62002
	2	0.1	0.57183	0.55661	0.55885	0.58069	0.58219	0.55825	0.56904	0.60582	0.572	0.55723	0.6926
		0.2	0.57814	0.56151	0.56307	0.57758	0.58472	0.55377	0.56828	0.55297	0.57243	0.59319	0.6762
		0.3	0.58139	0.56045	0.56453	0.57695	0.58613	0.55621	0.57678	0.57583	0.63517	0.62229	0.66291
		0.4	0.5733	0.56157	0.57201	0.57684	0.58162	0.54171	0.5714	0.5935	0.63969	0.61765	0.63683
		0.5	0.56818	0.56062	0.56647	0.569	0.5724	0.52849	0.56789	0.68637	0.67203	0.68307	0.61681

TABLE 4.10: Comparison of the proposed algorithm $HOPLP - MUL$ with baseline algorithms in terms of Balanced Accuracy Score layer-wise on three datasets and five *Ratio* values for testing to total edges percentage (contd..)

DATASET	Layer No.	Ratio	CN-WT	PA-WT	JC-WT	AA-WT	RA-WT	CC-WT	LOCALP-WT	NSILR-MUL	MADM-MUL	MVERSE-EMB	HOPLP-MUL
Kapferer-Tailor-Shop	3	0.1	0.58472	0.57213	0.58592	0.58854	0.59169	0.55019	0.5883	0.55734	0.77058	0.68336	0.65332
		0.2	0.58285	0.56722	0.58491	0.58881	0.57978	0.54079	0.58105	0.66678	0.68634	0.59994	0.63521
		0.3	0.57704	0.56792	0.58188	0.57669	0.57912	0.53942	0.58105	0.59198	0.63398	0.60371	0.61546
		0.4	0.56472	0.56328	0.57177	0.56591	0.56394	0.54365	0.56469	0.5756	0.60987	0.61947	0.60168
		0.5	0.55364	0.56389	0.54944	0.55198	0.55389	0.54634	0.55535	0.67141	0.65416	0.63408	0.56975
	4	0.1	0.59354	0.57091	0.59038	0.58852	0.59606	0.53106	0.59164	0.58152	0.50236	0.58209	0.65411
		0.2	0.59011	0.57465	0.59022	0.58578	0.58476	0.52807	0.59539	0.54792	0.58674	0.5838	0.63607
		0.3	0.57475	0.57473	0.57444	0.57855	0.57422	0.53687	0.57765	0.57568	0.61877	0.57447	0.6184
		0.4	0.56436	0.57345	0.56694	0.56437	0.56271	0.53096	0.57021	0.52745	0.59535	0.58233	0.61155
		0.5	0.55324	0.57142	0.55173	0.55582	0.55642	0.53195	0.56193	0.59971	0.59719	0.60605	0.57788
CKM-Physicians-Innovation	1	0.1	0.61021	0.54859	0.60956	0.60696	0.6117	0.46534	0.63944	0.82999	0.80232	0.66393	0.70775
		0.2	0.59206	0.54731	0.58903	0.59358	0.5935	0.46895	0.62449	0.81372	0.81003	0.65174	0.66641
		0.3	0.57663	0.54294	0.57341	0.57725	0.57581	0.47447	0.60437	0.8205	0.80408	0.6755	0.63834
		0.4	0.55777	0.5416	0.55709	0.55643	0.55712	0.48374	0.58381	0.8088	0.80479	0.69561	0.60788
		0.5	0.54318	0.53711	0.54135	0.54212	0.54183	0.49228	0.56265	0.80729	0.81356	0.72879	0.57939
	2	0.1	0.62437	0.51267	0.62149	0.62151	0.62569	0.45456	0.64853	0.69206	0.6716	0.65332	0.73352
		0.2	0.60547	0.51931	0.60368	0.60362	0.60306	0.45975	0.62405	0.67723	0.70542	0.63901	0.69925
		0.3	0.58381	0.51755	0.58398	0.58273	0.5833	0.46137	0.60766	0.68242	0.70507	0.67511	0.65814
		0.4	0.56361	0.51815	0.56394	0.56366	0.56421	0.4753	0.58425	0.68217	0.7117	0.68383	0.62089
		0.5	0.54593	0.51591	0.54694	0.54764	0.54799	0.4841	0.56332	0.72117	0.74452	0.69803	0.58733
	3	0.1	0.60555	0.51577	0.60115	0.59896	0.59892	0.48115	0.61855	0.50695	0.58152	0.567	0.69514
		0.2	0.58653	0.5152	0.58556	0.58553	0.58398	0.49088	0.60413	0.50876	0.58657	0.57339	0.65977
		0.3	0.56973	0.51436	0.56924	0.57122	0.57	0.49277	0.58833	0.53077	0.56639	0.59654	0.63023
		0.4	0.55435	0.50925	0.55324	0.5546	0.55335	0.49233	0.56946	0.54643	0.58107	0.61502	0.6036
		0.5	0.53834	0.50933	0.53935	0.53892	0.53903	0.49378	0.55107	0.58463	0.62221	0.63622	0.57385

$MADM-MUL$, and $HOPLP-MUL$ exhibit better performance when the Ratio variable is increased. $HOPLP-MUL$ algorithm performs better than $MVERSE-EMB$ for all layers of all datasets for $Ratio$ between 0.1&0.4.

The Balanced Accuracy score of the proposed $HOPLP-MUL$ algorithm is compared to that of baseline approaches in Tables 4.9 and 4.10. $HOPLP-MUL$ is the optimal weighted link prediction algorithm across all three datasets and all layers. Comparing $HOPLP-MUL$ approach to $NSILR-MUL$ and $MADM-MUL$, it is observed that this algorithm outperforms them for all $Ratio$ values between 0.1&0.4 in all layers except layer-1 of Vickers-Chan-7thGraders, layer-1,3 of Kapferer-Tailor-Shop, and layer-1,2 of CKM-Physicians-Innovation. As a result, it is observed that the method generates more false negatives than existing link prediction algorithms optimised for multiplex networks (for 0.5 probability threshold). Also it can be concluded that in cases where average shortest path length is greater than 2.5 $HOPLP-MUL$ algorithm's performance is worse than $MVERSE-EMB$ but overall complexity is much better. These tables only show networks in which the number of layers is less than five due to space constraints. These tables show that even after transforming the calculated likelihoods from the summarized weighted graph into the probability for edges on specific layers of the multiplex network, the proposed algorithm performs better than the classical link prediction algorithms.

4.4 Concluding Remarks

This chapter describes the $HOPLP-MUL$ technique for link prediction in multiplex networks based on the relevance of higher-order pathways and layer fusion. $HOPLP-MUL$ method sought to anticipate linkages by including more information about nodes (considerably larger zones of influence) and applying appropriate damping and layer fusion procedures. It uses an iterative approach to calculate link similarities over higher-order paths of the summarized graph. Even though longer paths are taken into account, we also account for more considerable resistance to information flow on such paths by using an adequate penalization approach that dampens the information flow from longer paths. The density-based proposed parameters and the modified initial significance play an essential role in the $HOPLP-MUL$ method. Using a layer-specific decompression constant, the link likelihoods on the summarized single-layer graph are utilized to forecast links on various layers of multiplex networks. This decompression constant is estimated using the difference in total densities across layers. The findings reveal that localized neighborhood-based algorithms have a relatively limited picture of the routes connecting nodes, resulting in reduced accuracy. This fact has been capitalized on in this chapter.